Remarks

The present amendment responds to the final Official Action dated April 19, 2005. The Official Action rejected claims 1-20 under 35 U.S.C. 103(a) as unpatentable over Bishop U.S. Patent No. 6,049,798 ("Bishop") in view of Barritz U.S. Patent No. 5,590,056 ("Barritz"), in view of Yamagishi U.S. Patent No. 5,870,604 ("Yamagishi") and in view of Farrell U.S. Patent No. 5,247,675 ("Farrell"). These grounds of rejection are addressed below after a brief description of the present invention to provide context.

The specification has been amended. Claims 1, 15 and 18 have been amended to be more clear and distinct. Claims 1-20 are presently pending.

The Present Invention

In one aspect, the present invention provides systems and techniques for recording information relating to operating software events as they occur. Various programs and tasks are controlled by the operating system, and performance, scheduling and other information relating to the programs and tasks may be recorded. The information may be maintained in a ledger where it is available for analysis in order to evaluate system performance. This analysis may be performed automatically or by a user, and the results of the analysis can be used to identify parameter changes that can be made in order to improve system performance.

Section 112 Rejections

The Official Action rejected claims 1-20 as indefinite. Claims 1, 15 and 18 have been amended to be more clear and distinct. With the present amendments, the Section 112 rejections have been overcome and should be withdrawn.

The Art Rejections

The art rejection hinges on the application of Bishop, Barritz, Yamagishi and Farrell, standing in combination. As addressed in greater detail below, Bishop, Barritz, Yamagishi and Farrell do not support the Official Action's reading of them and the rejections based thereupon should be reconsidered and withdrawn. Further, the Applicant does not acquiesce in the analysis of Bishop, Barritz, Yamagishi and Farrell made by the Official Action and respectfully traverses the Official Action's analysis underlying its rejections.

The Official Action rejected claims 1-20 under 35 U.S.C. 103(a) as unpatentable over Bishop, Barritz, Yamagishi and Farrell. In light of the present amendments to claims 1, 15 and 18, this ground of rejection is respectfully traversed.

Claim 1, as amended, claims that the operating software scheduling information capture software is operative to record a history of operating software events as they occur, information related to the history being organized and stored as operating software program scheduling information relating to interactions between the operating system software and each of the programs and tasks managed by the operating system software, the scheduling information including indications of relative priorities of programs and tasks, indications of transfers of control from lower priority to higher priority tasks and identification of tasks waiting for execution at the occurrence of each operating software event. Claim 1 further claims recording operating software scheduling information for relatively long duration storage in order to permit review of the scheduling information by a user. The duration of storage is required to be sufficient to allow data collected during an operating session to be retrieved and used after termination of the session.

This combination of features is not taught by Bishop. Bishop teaches systems and techniques for monitoring of resources in a data processing system. Bishop monitors the overall level of resource usage, for example CPU and RAM usage, without concentrating on which individual tasks are using which resource. Bishop monitors peripheral device usage by receiving information from device drivers. The device drivers sometimes provide information about the process using the device, but the information provided by device drivers, and the other resource usage information provided by Bishop, does not include a history of operating software events with information relating to the history being organized and stored as operating software program scheduling information relating to interactions between the operating system software and each of the programs and tasks managed by the operating system software, as is claimed by claim 1, as amended.

The operating software program scheduling information, as claimed by claim 1, as amended, provides a more elaborate picture of the interaction between the operating system and the various software components being managed by the operating system than does Bishop. In addition, the data collected by Bishop is not recorded for long duration storage in order to permit review of the scheduling information by a user, with the duration of storage being sufficient to allow data collected during an operating session to be retrieved and used after termination of the session.

Bishop is directed toward the display of resource utilization in real time and contemplates the monitoring and displaying of resource utilization information during an operating session, with no indication that the collected is to be preserved for retrieval or use beyond the session during which it is collected. Bishop's designation of a viewing period allows for a user of the system of Bishop to specify a time window for which information is to be presented, but any past

information collected for presentation within the time window is simply part of a continuous information stream, preserved so as to provide context for a user. This context allows a user to identify trends and changes and to see the effects changing conditions, including changes to the operation of the system caused by user actions and adjustments. Bishop does not indicate that the viewing period remains open past the termination of an operating session, and does not indicate that the collected information is retrievable past the termination of an operating session.

The storage of scheduling information claimed by claim 1 allows for retrieval and examination of information over a number of sessions. Such retrieval and examination allows for assembly and analysis of statistical information indicating system performance over a time period involving a number of sessions, and for comparison of system events and performance between different sessions. Analysis and comparisons made possible by preservation of scheduling information beyond the termination of a session allows for system refinements and improvements extending beyond those that can be performed during a single session. See specification p. 13, lines 17-23, which describes the retrieval of preserved information and notes the usefulness of such information in the development of robust embedded software.

Adding Barritz to Bishop does not cure Bishop's deficiencies as a reference with respect to claim 1, as amended. Claim 1 claims that a history of operating system of events organized and stored as scheduling information includes indications of relative priorities of programs and tasks, indications of transfers of control from lower priority to higher priority tasks and identification of tasks waiting for execution at the occurrence of each operating software event. Neither Bishop, Barritz nor a combination of Bishop and Barritz teaches or makes obvious these features. Bishop does not employ indications of relative priorities of programs and tasks. In order to track processor capacity that is not used by normally operating processes, Bishop begins

a new process that is given a lower priority than any executing process on the system. Bishop notes the time employed by this lowest priority process in order to determine time that is not used by any of the normal processes on the system. The lowest priority process can only run when none of the normal processes are running, so the run time of the lowest priority process can be tracked and used to determine the normal idle time for the system, that is, the time when none of the normal processes on the system need to run. Bishop uses this lowest priority process to determine the overall system usage, and does not track the relative priorities of different processes.

Adding Barritz to Bishop does not achieve or make obvious the invention as claimed by claim 1. Barritz teaches the collection of frequency of usage information for various software modules that may be hosted on a computer system. Barritz identifies a module when that module is invoked, identifying the particular software version represented by the module and matching module usage with licensing information in order to detect usage of unlicensed software. Barritz helps to identify underused and obsolete software modules in order to allow decisions to remove the underused and obsolete modules in order to save storage resources, and helps to identify unlicensed usage in order to assist in complying with licensing requirements. The information collected by Barritz does not include information relating to relative priorities of programs and tasks, transfers of control from lower priority to higher priority tasks and tasks waiting for execution at the occurrence of each operating software event. This information, collected by the present invention as claimed by claim 1, as amended, provides insight into the ongoing operation of a system and the allocation and management of resources to accomplish the various tasks performed by the system.

Adding Farrell to Bishop and Barritz does not teach or make obvious the invention as claimed by claim 1. Farrell teaches construction of threads having dispatch classes, and the storage and use of priority information in order to manage the assignment of priorities to threads and the allocation of processing resources among threads, but Farrell does not teach the relatively long term storage of scheduling information to permit review by a user, with the duration of storage being sufficient to allow data collected during an operating session to be retrieved and used after termination of the session. Storage of scheduling information to permit review by a user allows for insight by a user into the demands on and performance of a system and allows evaluation of the system to determine what adjustments, if any, need to be made, with the evaluation being able to take into account the demands and performance prevailing in multiple operating sessions and comparisons between demands and performance across multiple sessions. Such insights, and the relatively long term storage of information that permits the sort of review leading to such insights, are not provided by or relevant to Farrell, which simply uses priority information in operation but does not store it in order that it may be delivered for review by a user.

Adding Yamagishi to Bishop, Barritz and Farrell does not cure their deficiencies as references with respect to claim 1. Yamagishi distributes workloads among processors and uses data relating to the tasks awaiting execution and the number of tasks awaiting execution in order to manage workloads. Yamagishi does not collect data identifying the tasks awaiting execution at each particular software event for relatively long duration storage in order to permit review by a user, but simply uses the information in operation. Claim 1 claims recording scheduling information that includes identification of tasks awaiting execution at each software event so that the scheduling information may be presented to a user for review. Such collection and

presentation of information allows for insight into the operation of a system and helps to determine what adjustments, if any, need to be made. The relatively long term storage for presentation to a user is not accomplished by or relevant to Yamagishi, which simply uses information relating to the tasks or numbers of task awaiting execution in workload balancing, but does not contemplate that the information will be reviewed in order to provide insight by a user into the operation of the system. Claim 1, as amended, therefore defines over the cited art and should be allowed.

Claim 15, as amended, claims invoking operating software scheduling information capture software. The operating software scheduling information capture software is operative to record a history of operating software events as they occur. The history is organized and stored as operating software program scheduling information relating to interactions between the operating system software and each of the programs and tasks managed by the operating system software. The scheduling information includes indications of relative priorities of programs and tasks, indications of transfers of control from lower priority to higher priority tasks and identification of tasks waiting for execution at the occurrence of each operating software event. Claim 15 further claims recording operating software scheduling information for relatively long duration storage in order to permit review of the scheduling information by a user. Duration of storage is sufficient to allow data collected during an operating session to be retrieved and used after termination of the session. For the reasons stated above with respect to claim 1, neither Bishop, Barritz, Farrell, Yamagishi nor a combination thereof teaches or makes obvious these limitations. Claim 15, as amended, therefore defines over the cited art and should be allowed.

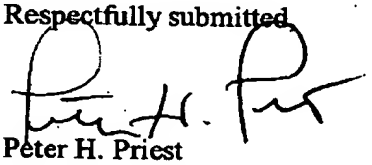
Claim 18, as amended, claims a memory coupled to a processor, the memory having stored therein sequences of instructions which, when executed by the processor, cause the

processor to invoke operating software scheduling information capture software. The operating software scheduling information capture software is operative to record a history of operating software events as they occur. The history is organized and stored as operating software program scheduling information relating to interactions between the operating system software and each of the programs and tasks managed by the operating system software. The scheduling information includes indications of relative priorities of programs and tasks, indications of transfers of control from lower priority to higher priority tasks and identification of tasks waiting for execution at the occurrence of each operating software event. Claim 18 further claims recording operating software scheduling information for relatively long duration storage in order to permit review of the scheduling information by a user. Duration of storage is sufficient to allow data collected during an operating session to be retrieved and used after termination of the session. For the reasons stated above with respect to claim 1, neither Bishop, Barritz, Farrell, Yamagishi nor a combination thereof teaches or makes obvious these limitations. Claim 18, as amended, therefore defines over the cited art and should be allowed.

Conclusion

All of the presently pending claims, as amended, appearing to define over the applied references, withdrawal of the present rejection and prompt allowance are requested.

Respectfully submitted,



Peter H. Priest
Reg. No. 30,210
Priest and Goldstein, PLLC
5015 Southpark Drive, Suite 230
Durham, NC 27713-7736
(919)806-1600